A New Interactive Hashing Theorem

Iftach Haitner
Dept. of Computer Science and Applied Math.
Weizmann Institute of Science
Rehovot 76100, Israel
iftach.haitner@weizmann.ac.il

Omer Reingold
Dept. of Computer Science and Applied Math.
Weizmann Institute of Science
Rehovot 76100, Israel
omer.reingold@weizmann.ac.il

Abstract

Interactive hashing, introduced by Naor, Ostrovsky, Venkatesan and Yung (CRYPTO '92), plays an important role in many cryptographic protocols. In particular, it is a major component in all known constructions of statistically-hiding commitment schemes and of zero-knowledge arguments based on general one-way permutations and on one-way functions. Interactive hashing with respect to a one-way permutation f, is a two-party protocol that enables a sender that knows $y = f(x)$ to transfer a random hash $z = h(y)$ to a receiver. The receiver is guaranteed that the sender is committed to y (in the sense that it cannot come up with x and x' such that $f(x) \neq f(x')$ but $h(f(x)) = h(f(x')) = z$). The sender is guaranteed that the receiver does not learn any additional information on y. In particular, when h is a two-to-one hash function, the receiver does not learn which of the two preimages $\{y, y'\} = h^{-1}(z)$ is the one the sender can invert with respect to f.

This paper reexamines the notion of interactive hashing. We give an alternative proof for the Naor et. al. protocol, which seems to us significantly simpler and more intuitive than the original one. Moreover, the new proof achieves much better parameters (in terms of how security preserving the reduction is). Finally, our proof implies a more versatile interactive hashing theorem for a more general setting than that of the Naor et. al. protocol. One generalization relates to the selection of hash function h (allowing much more flexibility). More importantly, the theorem applies to the case where the underlying function f is hard-to-invert only on some given (possibly sparse) subset of the output strings. In other words, the theorem is tuned towards hashing of a value y that may be distributed over a sparse subset of the domain (rather than uniform on the entire domain as a random output of a one-way permutation is).

Our interest in interactive hashing is in part as a very appealing object (i.e., independent of any particular application). Furthermore, a major motivation for looking into interactive hashing is towards improving and simplifying previous constructions of statistical zero-knowledge and statistical commitments (that employ interactive hashing as a central building block). We make some preliminary progress in this direction as well.

1 Introduction

Interactive hashing, introduced by Naor, Ostrovsky, Venkatesan and Yung [12], is a protocol that allows a sender $\mathcal{S}$ to commit to a particular value while only reviling to a receiver $\mathcal{R}$ some predefined information of this value. More specifically, $\mathcal{S}$ commits to a value y while only reviling to $\mathcal{R}$ the value $(h, h(y))$, where h is some random hash function (we defer additional details on the choice of hash function). The two security properties of interactive hashing are *binding* (namely, $\mathcal{S}$ is bounded by the protocol to at most one value of y) and *hiding* (namely, $\mathcal{R}$ does not learn any impermissible information about y). As in [12], we will consider in this paper interactive hashing where the hiding property is statistical (i.e., the protocol preserves the secrecy of y even against an all-powerful $\mathcal{R}$), and the binding property is computational (i.e., it assumes that S is computationally bounded).

Interactive hashing (in the flavor mentioned above) is closely related and to a large extent motivated by the fundamental notion of statistical commitments (i.e., statistically-hiding computationally-binding commitment schemes). In statistical commitments, we again have a protocol between a sender $\mathcal{S}$ and a receiver $\mathcal{R}$. However, here the sender $\mathcal{S}$ commits to y without reviling *any* information about y. Statistical commitments can be used as a building block in constructions of statistical zero-knowledge arguments [1, 12] or certain coin-tossing protocols [11]. More generally, they have the following advantage over computationally hiding commitment schemes when used within protocols in which certain commitments are never revealed: in such a scenario, it need only be infeasible to violate the binding property *during the period of time the protocol is run*, whereas the committed values will remain hidden *forever* (i.e., regardless of how much time the receiver invests after completion of the protocol).

The relation between interactive hashing and statistical commitments goes beyond the similarity in definitions. On one hand, interactive hashing can easily be implemented using commitment schemes (simply commit to y using the commitment scheme and reveal whatever information needed on y in the clear). On the other hand, one of the main applications of interactive hashing protocols is for constructing statistical commitment schemes. Indeed, interactive hashing is a major component in all known constructions (listed below) of statistical commitment that are based on, possibly restricted types of, one-way functions.

Naor et. al. [12] use their interactive hashing protocol (from now on the NOVY protocol) in order to construct statistical commitments based on any one-way permutation. Haitner et. al. [7] make progress by using the NOVY protocol to construct statistical commitments based on regular one-way functions and also on the so called approximable-size one-way functions. Recently, Haitner and Reingold [8] constructed statistical commitments based on *any* one-way functions. Their construction makes use of two-phase commitment schemes, recently introduced and implemented by Nguyen et al. [13] in their breakthrough construction of statistical zero-knowledge arguments for NP based on any one-way function. Not surprisingly, interactive hashing is heavily used in the [13] implementation of two-phase commitment schemes.

Interactive hashing is also used by several other cryptographic protocols [15, 16, 17]. In addition, it is used in "information theoretic setting" (i.e., no hardness assumptions are assumed) such as [2, 4, 5, 14].

A possible drawback of [13] (and therefore also of [8]) is that the construction is rather inefficient and complex. Indeed, a major motivation for looking into interactive hashing is to simplify constructions of statistical zero-knowledge arguments for NP and constructions of statistical commitment schemes based on any one-way functions. However, before discussing our results and their applications let us have a closer look into the notion of interactive hashing.

1.1 Interactive Hashing in the Setting of One-Way Permutations

Consider the following two-party protocol between a sender $\mathcal{S}$ and a receiver $\mathcal{R}$: The sender chooses a random element $x \in \{0,1\}^n$ and sets $y = f(x)$, where $f : \{0,1\}^n \mapsto \{0,1\}^n$ is a one-way permutation. Next, the receiver selects a random two-to-one hash function $h : \{0,1\}^n \mapsto \{0,1\}^{n-1}$ and sends its description to $\mathcal{S}$. Finally, $\mathcal{S}$ sends $z = h(y)$ back to $\mathcal{R}$. Note that if both parties follow the protocol, then the following "binding" property is guaranteed: It is not feasible for $\mathcal{S}$ to find a *second* element $x' \in \{0,1\}^n$ such that $f(x') \neq f(x)$ but $h(f(x')) = h(f(x)) = z$, although (exactly one) such element x' does exist. The reason is that the task of finding such x' can easily be shown to be equivalent in hardness to inverting f on a random output element (the latter task is assumed to be hard by the one-wayness of f). Furthermore, we are guaranteed to have the following "hiding" property: Let y_1 and y_2 be the two preimages of z w.r.t. h. Given R's view of the communication (i.e., given the values h and z), it is indistinguishable whether the random element chosen by $\mathcal{S}$ is $x_1 = f^{-1}(y_1)$ or $x_2 = f^{-1}(y_2)$. In this sense, $\mathcal{S}$ has committed to a bit (which indicates if it can produce the inverse of y_1 or that of y_2). This bit is statistically hidden from $\mathcal{R}$.

What happens, however, if $\mathcal{S}$ selects x only *after* seeing h? In such a case, it is quite plausible that $\mathcal{S}$ would be able to "cheat" by producing $x, x' \in \{0,1\}^n$ such that $f(x) \neq f(x')$ but $h(f(x')) = h(f(x)) = z$.[1] The NOVY interactive hashing protocol prevents exactly such cheating. For that it employs a specific family of hash functions such that each one of its functions h can be decomposed into $n-1$ Boolean functions $h_1, \ldots, h_{n-1}$, where $h(x) = h_1(x), \ldots, h_{n-1}(x)$.[2] In the NOVY protocol, instead of sending h at once as described above, $\mathcal{R}$ sends a single Boolean function h_i in each round. In return, the honest sender sends a bit $z_i = h_i(f(x))$. What about a cheating sender? Intuitively, a cheating sender has a significantly smaller leeway for cheating as it can no longer wait in selecting x till it receives the entire description of h. Still, it is highly non-trivial to argue (formally or even intuitively) that restricting the sender by adding interaction in this manner is sufficient in order to prevent the sender from cheating. Perhaps surprisingly, Naor et. al. [12] have shown that their protocol has the binding property even against a cheating sender (namely, even a cheating sender cannot produce $x, x' \in \{0,1\}^n$ such that $f(x) \neq f(x')$, but $h(f(x')) = h(f(x)) = z$).

1.2 Interactive Hashing in the Sparse Case

The NOVY interactive hashing protocol applies to one-way permutations and easily implies the existence of statistical commitments from any one-way permutation. [3] How

[1] Assume for example that the one-way permutation equals the identity function on the set T of all strings that start with $n/4$ zeros (where n is the input length). Now given a hash function h all the cheating sender has to do is to find a collision $y_1 \neq y_2$, where $y_1, y_2 \in T$, such that $h(y_1) = h(y_2)$. Such a collision is likely to exist by the birthday paradox, and for many families of hash functions finding such a collision is easy.

[2] For more details on the definition of this family of hash functions see Section 4.

[3] Actually, the NOVY commitment schemes are even stronger being perfectly hiding.

about constructing statistical commitments from, say, regular one-way functions (one-way functions where every output value has the same number of preimages)? In such a case we would like to interactively hash a value y (a random output of the one-way function) which is uniformly distributed in some subset L of $\{0,1\}^n$ (rather than uniformly distributed in all of $\{0,1\}^n$ as in the case of one-way permutations). What is the difficulty in directly hashing a value y that is taken from a set L that is sparse in $\{0,1\}^n$? The NOVY-theorem guarantees that when hashing y with $h : \{0,1\}^n \mapsto \{0,1\}^{n-1}$ the sender is committed to a single value y (as shown in [13] this holds even if the output of h is a bit shorter). However, when h outputs so many bits then most likely $h(y)$ completely determines y and statistical hiding is lost.

Facing the aforementioned difficulty, Haitner et. al. [7] first make the following observation: the NOVY protocol is still meaningful even when hashing a value y which is taken from a distribution that is "dense" in $\{0,1\}^n$ (a bit more formally we would like the distribution to be sufficiently close to having min-entropy $n - O(\log n)$). In particular, if the one-way function is length-preserving poly-to-one (i.e., each output has at most polynomial number of preimages in the image set of f), then the NOVY-protocol can be applied as is to give some weak form of statistical commitments that can later be amplified to full-fledge statistical commitments. To handle any regular one-way function, [7] applies additional layer of (non interactive) hashing to reduce to the dense case. This implies a construction of statistical commitments from any regular one-way function with known image size. Interactive hashing in the sparse case arises in other works as well, most notably in the construction of statistical zero-knowledge arguments from any one-way function [13].

1.3 Our Results

We introduce an alternative proof for the NOVY protocol, which relies in parts on the original proof due to [12] (the NOVY proof) but still seems to us significantly simpler. The proof follows a simple intuition that is sketched below in this section. Moreover, the parameters achieved by our proof are an improvement compared with the original ones. In our proof, given an algorithm A that breaks the binding property with probability ε_A we get an algorithm that inverts the one-way permutation in comparable time and with inverting probability $\varepsilon_A^2 \cdot \mathsf{poly}(n)$. This is a substantial improvement and is much closer to natural limitations of the proof technique (see discussion in Section 5). [4]

In addition to being simpler and more security preserving, the new proof implies a more general interactive hashing theorem. The new theorem applies to every family of hash functions that is a product of Boolean families of pairwise independent hash functions (and not only to the special family of two-to-one hash functions used by [12, 13]). More importantly, the new theorem directly applies to the "sparse case": Let $f : \{0,1\}^n \mapsto \{0,1\}^{\ell(n)}$ be an efficiently computable function, let $L \subseteq \{0,1\}^{\ell(n)}$. As mentioned above, when hashing a value $y \in L$, the NOVY proof only promises binding when using a hash function that outputs almost n bits. However, in such a case y is likely to be completely determined by $h(y)$ and statistical hiding cannot be guaranteed. Our theorem applies even when hashing to roughly $\lfloor \log(|L|) \rfloor$ bits. In particular, when h is taken from a family of hash functions $\overline{\mathcal{H}} : \{0,1\}^{\ell(n)} \mapsto \{0,1\}^{\lfloor \log(|L|) \rfloor}$ that is a product of $\lfloor \log(|L|) \rfloor$ families of pairwise independent Boolean hash functions, we can show that a close variant of the NOVY protocol possesses the following binding property: If f is hard to invert on the uniform distribution over L, then no polynomially-bounded sender $\mathcal{S}^*$ (even one which arbitrarily deviates from the protocol) can find two elements $x, x' \in f^{-1}(L)$ such that $f(x) \neq f(x')$ but $h(f(x)) = h(f(x')) = z$ (where z is the value determined by the protocol as $h(y)$).

1.4 Applications of the New Theorem

In the full version, we use the new theorem to derive a direct construction of statistical commitment based on known regular one-way functions (and thus reprove [7]). We also believe that our new result can also used to simply the construction of two-phase commitment schemes given in [13]. Thus, it can simplify both the [13] construction of statistical zero-knowledge arguments for NP and the [8] construction of statistical commitments.

1.5 Related Work

We note that independently of our work, Nguyen et al. [13] give a new proof for the NOVY protocol. Their proof follows the proof of [12] more closely than ours but still introduces various simplifications and parameter improvements. The main goal of the new proof is to generalize the protocol such that it allows hashing with a hash function that is poly-to-one rather than two-to-one as in [12]. In other words, they analyze the NOVY protocol with $n - \ell$ rounds where ℓ may be as large as $O(\log n)$ rather than $n-1$ rounds in [12]. For a comparison between the parameters obtained by [12], [13] and this paper, see Remark 3.11.

Wee [18] have recently showed that large classes of (fully) black-box construction of interactive hashing based on one-way permutations requires $\Omega(n/log(n))$ rounds. [5]

[4] We note that independently of our work, [13] recently presented an $\varepsilon_A^3 \cdot \mathsf{poly}(n)$ reduction. See discussion below for more detail on their work.

[5] Informally, a protocol in this class first invokes f on randomly chosen inputs and then it cannot access the inputs any longer.

In this paper we give a protocol that uses $\theta(n)$ rounds (i.e., same as in [12, 13]). It is easy, however, to extend our analysis to a variant of our protocol that uses hash functions with output length $\theta(\log(n))$ rather than Boolean hash functions. Therefore, the number of rounds in this modified protocol would be $\theta(n/log(n))$. This was also recently shown for a similar modification of the NOVY analysis [10]. Thus, w.r.t. black-box construction our protocol is tight in this sense.

1.6 Relations

Following [13], we state our protocol, and proof, in the more general setting of binary relations rather than functions. For instance, given a binary relation W that is hard to satisfy (i.e., given y it is hard to find x such that $(x, y) \in W$), we prove that following our interactive hashing protocol, $\mathcal{S}$ cannot find two pairs $(x_0, y_0), (x_1, y_1) \in W$ such that both y_0 and y_1 are consistent with the protocol, but $y_0 \neq y_1$. Note that since every function, f, defines the natural binary relation $(x, f(x))$, any result w.r.t. binary relations implies an equivalent result w.r.t. functions.

1.7 Cheating Receiver

In our new interactive hashing theorem, the hiding property of a cheating receiver is specified only with respect to a semi-honest receiver (a.k.a. honest-but-curious). A malicious receiver can learn $h(y)$ where h is not necessarily uniformly distributed. In fact, h can be determined adaptively based on partial knowledge of $h(y)$ (specifically, h_i is selected after learning the first $i - 1$ bits of $h(y)$). The NOVY protocol provides a cheating receiver exactly the same power in selecting h. Nevertheless, when y is selected uniformly in $\{0, 1\}^n$ and h is two-to-one, then regardless of how the receiver selects h there is one bit of knowledge on y that remains completely hidden from the receiver. That is, given $z = h(y)$ there are two possibilities to the value of y, the hidden bit specifies which of these two values is the right one. In the general case, when y is distributed over a sparse subset, one should take more care in estimating the power of a cheating receiver. We note that assuming the existence of one-way functions, in various settings, one can assume without loss of generality that the receiver is semi-honest. In particular, this is the case for statistical commitments ([7]).

1.8 Proof Idea

We discuss our binding proof in the most basic setting where $f : \{0, 1\}^n \mapsto \{0, 1\}^n$ is a one-way permutation and $L = \{0, 1\}^n$ (the proof of general case is not significantly different). Our protocol consists of $m = n - \mathcal{O}(\log(n))$ rounds. First, $\mathcal{S}$ selects a random element $x \in \{0, 1\}^n$, then in each round, $\mathcal{R}$ selects a random Boolean pairwise-independent hash function h_i and $\mathcal{S}$ replies with $z_i = h_i(f(x))$.

Let A be an algorithm that plays the sender's role in the protocol and at the end of the protocol outputs two elements $x_1, x_2 \in \{0, 1\}^n$. Assume that with some noticeable probability ε, it holds that $f(x_1) \neq f(x_2) \in \mathsf{Consist}(h_1, \ldots, h_m)$, where $\mathsf{Consist}(h_1, \ldots, h_m) \overset{\text{def}}{=} \{y \in \{0, 1\}^n : \forall i \in [m] \;\; h_i(y) = z_i\}$. It is easy to use A in order to construct an algorithm that inverts f with probability $\frac{\varepsilon}{2^n}$: Given input y, the algorithm chooses the hash functions at random and returns one of the two values that A outputs. [6]

Let's imagine that instead we are trying to invert f on the following distribution: The first $k = m - \log(\frac{1}{\varepsilon}) - C\log(n)$ (for some constant $C > 0$ determined by the analysis) Boolean hash functions, $h_1, \ldots, h_k$, are chosen at random and only then a random element, y, is uniformly drawn from $\mathsf{Consist}(h_1, \ldots, h_k)$. We call the distribution induced on $(y, h_1, \ldots, h_k)$ by the above process D_{Ideal}. On the average, A has probability ε to cheat even when conditioned on $h_1, \ldots, h_k$ being selected. Also note that by the pairwise independence of the h's, the size of $\mathsf{Consist}(h_1, \ldots, h_k)$ is, with high enough probability, about $\frac{2^n}{2^{m-k}} = \frac{n^C}{\varepsilon}$. It follows that in this setting the naive algorithm, which selects the rest of the hash functions at random and returns one of A's answers, inverts f with probability close to $\frac{\varepsilon^2}{n^C}$. [7]

Let's try to emulate the above setting on a random $y \in \{0, 1\}^n$. Namely, given a uniformly chosen $y \in \{0, 1\}^n$, we will choose $h_1, \ldots, h_k$ so that $(y, h_1, \ldots, h_k)$ will have about the same distribution as it was drawn from D_{Ideal}. To do so, we choose $h_1, \ldots, h_k$ one by one, each time we keep sampling until we find an hash function that its value on y is consistent with A's answer (if the answer is inconsistent, we "rewind" A to its state before it was asked the last "faulty" hash function). We call D_{Real} the distribution the above process induces on $(y, h_1, \ldots, h_k)$. We would conclude the proof of the binding property if we could prove that the statistical difference between D_{Ideal} and D_{Real} is smaller than $\frac{\varepsilon^2}{n^C}$ (recall that this is the inverting probability of the naive algorithm on D_{Ideal}). Unfortunately, we cannot prove such a strong bound.

Note that till now we did not take advantage of the full powers of A, since we did not use the fact that A finds two different outputs of f that are consistent with the protocol and not merely a single one (indeed the above observations

[6] With probability $\frac{|\mathsf{Consist}(h_1,\ldots,h_m)|}{2^n}$, a random y is uniformly distributed in $\mathsf{Consist}(h_1, \ldots, h_m)$. Now if $f(x_1)$ or $f(x_2)$ are in $\mathsf{Consist}(h_1, \ldots, h_m)$ (which clearly happens with probability at least ε), then with probability at least $\frac{1}{|\mathsf{Consist}(h_1,\ldots,h_m)|}$ either x_1 or x_2 are the inverse of y. All in all, we invert y with probability $\frac{|\mathsf{Consist}(\mathcal{H}_1,\ldots,h_m)|}{2^n} \cdot \varepsilon \cdot \frac{1}{|\mathsf{Consist}(h_1,\ldots,h_m)|} = \frac{\varepsilon}{2^n}$.

[7] same argument as in the above case where y is uniformly chosen from $\{0, 1\}^n$.

hold also w.r.t. the honest $\mathcal{S}$). When taking into account the full powers of A, we manage to prove that the success probability of the naive algorithm w.r.t. D_{Ideal} does not depend on inverting too few elements. More specifically, the subset of $y \in \mathsf{Consist}(h_1, \ldots, h_k)$ such that the naive algorithm inverts on them with "high enough" probability is of relative size $\sqrt{\varepsilon \cdot |\mathsf{Consist}(h_1, \ldots, h_k)|}$. [8]

The latter observation turns to be useful, since we also mange to prove the following. For most choices of $h_1, \ldots, h_k$ (excluding a set of probability much smaller than $\frac{\varepsilon^2}{n^C}$ - the success probability of the naive algorithm, described above, w.r.t. D_{Ideal}), for most elements in $\mathsf{Consist}(h_1, \ldots, h_k)$ (excluding a set of size much smaller than $\sqrt{\varepsilon \cdot |\mathsf{Consist}(h_1, \ldots, h_k)|}$) the probability mass that $(y, h_1, \ldots, h_k)$ has under D_{Ideal} is within a constant factor from its mass under D_{Real}. By the above observations, it follows that we can invert y with noticeable probability over D_{Real}, which directly implies that we can invert f (again, with noticeable probability) on the uniform distribution over $\{0,1\}^n$.

1.9 Paper Organization

In Section 3, we generalize the definition of interactive hashing, present our new construction and prove that it satisfies the new definition. In Section 4, we argue that the new proof can also be applied to the original NOVY protocol (that uses very specific hash functions). Discussion and further issues appear in Section 5.

2 Preliminaries

2.1 Notation

For $k \in \mathbb{N}$, we denote by $[k]$ the set $\{1, \ldots, k\}$. We denote the concatenation of the strings x and y by $x \circ y$. Given a set L, we denote by $x \leftarrow L$ the experiment in which x is uniformly chosen from L. Let D be a distribution over the set L, the support of D is defined as: $sup(D) \stackrel{\text{def}}{=} \{x \in L : D(x) > 0\}$. We denote the probability of $L' \subseteq L$ w.r.t. D as $D(L') \stackrel{\text{def}}{=} \Pr_{x \leftarrow D}[x \in L']$. Given a function $f : \{0,1\}^* \mapsto \{0,1\}^*$ and a set $L \subseteq \{0,1\}^*$ and denote the image of f on L as $f(L) \stackrel{\text{def}}{=} \{f(x) : x \in L\}$ and denote $f(\{0,1\}^*)$ by $Im(f)$. For $y \in Im(f)$, we define $f^{-1}(y) \stackrel{\text{def}}{=} \{x \in \{0,1\}^* : f(x) = y\}$. The statistical distance of two distributions P and Q over Ω, denoted $SD(P, Q)$, is defined as

$$SD(P, Q) \stackrel{\text{def}}{=} \frac{1}{2} \sum_{x \in \Omega} \left| \Pr_P(x) - \Pr_Q(x) \right| \ .$$

We denote the running time of an algorithm A by T_A, where PPT stands for polynomial-time algorithm. Given two interactive Turing machines (ITM) A and B, we denote the protocol they define by (A, B) and denote the following experiment by $(o_A, o_B) \leftarrow \langle A(i_A), B(i_B) \rangle$: The protocol (A, B) is invoked with inputs i_A and i_B and the outputs of the parties are assigned to o_A and o_B respectively.

2.2 Pairwise Independent Hash Functions

Definition 2.1 (pairwise independent hash functions)**.** *Let $\mathcal{H}$ be a family of functions mapping strings of length $\ell(n)$ to strings of length $m(n)$. We say that $\mathcal{H}$ is an* efficient family of pairwise independent hash functions *(following [3]) if the following hold:* [9]

Samplable. *$\mathcal{H}$ is polynomially samplable (in n).*

Efficient. *There exists a polynomial-time algorithm that given $x \in \{0,1\}^{\ell(n)}$ and a description of $h \in \mathcal{H}$ outputs $h(x)$.*

Pairwise independence. *For every distinct $x_1, x_2 \in \{0,1\}^{\ell(n)}$ and every $y_1, y_2 \in \{0,1\}^{m(n)}$, we have:*

$$\Pr_{h \leftarrow \mathcal{H}}[h(x_1) = y_1 \bigwedge h(x_2) = y_2] = 2^{-2m(n)} \ .$$

It is well known ([3]) that there exists an efficient family of pairwise-independent hash functions for every choice of ℓ and m whose elements description size is $\mathcal{O}(\max\{\ell(n), m(n)\})$.

The following standard lemma (see for example, [6, Lemma 4.3.1]) states that a random pairwise independent hash function partitions a given set into (almost) equal size subsets.

Lemma 2.2. *Let $\mathcal{H}$ be a family of pairwise independent hash functions mapping strings of length ℓ to strings of length m, let $L \subseteq \{0,1\}^\ell$ and let $\mu = \frac{|L|}{2^m}$. Then for every $y \in \{0,1\}^m$ and $\delta > 0$*

$$\Pr_{h \leftarrow \mathcal{H}}[||\{x \in L : h(x) = y\}| - \mu| > \delta\mu] < \tfrac{1}{\delta^2 \mu} \ .$$

As in [12], our interactive hashing protocol selects and evaluates an hash function incrementally. Therefore, the protocol is designed for hash functions that are product of hash functions defined next.

[8] Loosely, let T be the set of y's that A is likely to output their inverse (according to f). A random selection of $h_{k+1}, \ldots, h_m$ separates every two elements in T with probability $1 - 2^{-(m-k)}$. So unless the size of T is large enough, one of the two values A output will be forced to be the inverse of an element outside of T. This will contradict the assumptions that values outside of T are only inverted with small probability.

[9] The first two properties, regarding the efficiency of the family, implicitly assume an ensemble of families (one family for every value of n). For simplify of presentation, we only refer to a single family.

Definition 2.3 (product hash family)**.** *Let $\mathcal{H}$ be a family of functions mapping strings of length $\ell(n)$ to strings of length $m(n)$ and let $k(n) \in \mathbb{N}$. The k*-product-family *of $\overline{\mathcal{H}}$, denoted $\mathcal{H}^{\times k(n)}$, is a family of functions mapping strings of length $\ell(n)$ to strings of length $k(n) \cdot m(n)$ which is defined as follows: The members of $\overline{\mathcal{H}}$ are all possible tuples $\overline{h}$ of $k(n)$ functions from $\mathcal{H}$. For every such tuple $\overline{h} = (\overline{h}_1, \ldots, \overline{h}_{k(n)})$ and every $x \in \{0,1\}^{\ell(n)}$ we define $\overline{h}(x) = (\overline{h}_1(x), \ldots, \overline{h}_{k(n)}(x))$.*

3 The New Interactive Hashing Theorem

In this section we present our extended definition for an interactive hashing protocol and give a revised construction and new proof that match this definition.

3.1 Defining a New Notion of Interactive Hashing

We choose (following [13]) to state our definitions in the setting of binary relations. This generalizes the original definition due to [12], which concentrates on the particular relations that are naturally defined by one-way permutations (see Corollary 3.15). In particular, the underlying relation is not necessarily efficiently computable or even not efficiently verifiable. Moreover, the relation is not necessarily defined over all strings of a given length, but might rather be defined over some small subset of the strings.

Notation: Let W be a binary relation and let $y \in \{0,1\}^*$, we denote the set $\{x \in \{0,1\}^* : W(x,y) = 1\}$ by W_y.

Definition 3.1 (interactive hashing)**.** *Let $\overline{\mathcal{H}}$ be a family of hash functions mapping strings of length $\ell(n)$ to strings of length $m(n)$. An $\overline{\mathcal{H}}$-interactive hashing $(\mathcal{S}, \mathcal{R})$, with security parameter n, is a probabilistic polynomial-time interactive protocol. Both parties receive the security parameter 1^n as an input and $\mathcal{S}$ gets as a private input $y \in \{0,1\}^{\ell(n)}$. At the end, $\mathcal{S}$ locally outputs y and $\mathcal{R}$ outputs $(\overline{h}, \overline{z}) \in \overline{\mathcal{H}} \times \{0,1\}^{m(n)}$.*

We make the following correctness requirement: For all n, all $y \in \{0,1\}^{\ell(n)}$, and every pair $(y, (\overline{h}, \overline{z}))$ that may be output by $\langle \mathcal{S}(1^n, y), \mathcal{R}(1^n) \rangle$, it is the case that $\overline{h}(y) = \overline{z}$.

The security of interactive hashing protocol has two aspects. Binding the sender to y and concealing some information regarding y from $\mathcal{R}$. In this paper we focus on security w.r.t. polynomially-bounded sender and unbounded receiver. The setting where both the receiver and the sender are unbounded, called Information Theoretic interactive hashing (a.k.a. interactive hashing for static sets), is not treated by this paper (for details on the information theoretic setting see for example [2, 4, 5]). We start by formalizing the binding property.

Definition 3.2 ($\mathsf{BindBreak}^{\mathsf{L,W}}$)**.** *Let $L \subseteq \{0,1\}^{\ell(n)}$, let W be a binary relation and let $\overline{\mathcal{H}}$ be a family of hash functions mapping strings of length $\ell(n)$ to strings of length $m(n)$. Let $o_{\mathcal{S}} = ((x_0, y_0), (x_1, y_1)) \in (\{0,1\}^* \times \{0,1\}^{\ell(n)})^2$ and let $o_{\mathcal{R}} = (\overline{h}, \overline{z}) \in \mathcal{H} \times \{0,1\}^{m(n)}$. We define the predicate*

$$\mathsf{BindBreak}^{\mathsf{L,W}}(o_{\mathcal{S}}, o_{\mathcal{R}}) \overset{def}{=} y_0 \neq y_1 \in L$$
$$\bigwedge \overline{h}(y_0) = \overline{h}(y_1) = \overline{z} \bigwedge (x_0, y_0), (x_1, y_1) \in W \enspace .$$

Definition 3.3 (binding)**.** *Let L, W and $\overline{\mathcal{H}}$ be as in Definition 3.2 and let $(\mathcal{S}, \mathcal{R})$ be an $\overline{\mathcal{H}}$-interactive-hashing protocol. We say that $(\mathcal{S}, \mathcal{R})$ is* binding *w.r.t. L and W if for every* PPT *A that plays the role of $\mathcal{S}$ in the protocol and outputs two pairs of elements, the following is negligible:*

$$\Pr_{(o_A, o_{\mathcal{R}}) \leftarrow \langle A(1^n), \mathcal{R}(1^n) \rangle}[\mathsf{BindBreak}^{\mathsf{L,W}}(o_A, o_{\mathcal{R}})] \enspace ,$$

where the probability is taken over the random coins of A and $\mathcal{R}$.

Remark 3.4. *Comparing to NOVY: The binding proof of Naor et. al. [12] holds w.r.t. $L = \{0,1\}^n$, W is the relation naturally defined by a one-way permutation and $\overline{\mathcal{H}}$ is a specific type family of two-to-one Boolean hash functions. See Section 4 for more details on the NOVY protocol.*

The following definition states that the only information that a semi-honest receiver acquires through the protocol about y is its hash value for a uniformly chosen hash function.

Definition 3.5 (secrecy preserving w.r.t. semi-honest receiver)**.** *Let $\overline{\mathcal{H}}$ be a family of hash functions mapping strings of length $\ell(n)$ to strings of length $m(n)$ and let $(\mathcal{S}, \mathcal{R})$ be an $\overline{\mathcal{H}}$-interactive-hashing protocol. For $y \in \{0,1\}^{\ell(n)}$, we denote by $\mathsf{view}_{\langle \mathcal{S}(y), \mathcal{R} \rangle}(n)$ the distribution of the view of $\mathcal{R}(1^n)$ when interacting with $\mathcal{S}(1^n, y)$ (this view simply consists of the sequence of messages $\mathcal{R}$ receives from $\mathcal{S}$ and its random coins), where this distribution is taken over the random coins of $\mathcal{S}$ and $\mathcal{R}$. We say that $(\mathcal{S}, \mathcal{R})$ is* secrecy-preserving (w.r.t. semi-honest receiver) *if there exists a polynomial-time simulator Sim, such that for every $y \in \{0,1\}^{\ell(n)}$ the distributions $\mathsf{view}_{\langle \mathcal{S}(y), \mathcal{R} \rangle}(n)$ and $\big(Sim(1^n, \overline{h}, \overline{h}(y))\big)_{\overline{h} \leftarrow \overline{\mathcal{H}}}$ are identical.*

Remark 3.6. *Some level of hiding can be guaranteed by our protocol even against* malicious *$\mathcal{R}$. Specifically, the protocol hides any information regarding the index of y among all the preimages of $\overline{z} = \overline{h}(y)$ w.r.t. $\overline{h}$. In the setting of [12] this information is quite meaningful and is also easy to construct. This is because y is chosen uniformly in $\{0,1\}^{\ell(n)}$ and regardless of the way the receiver selects $\overline{h}$, there are exactly two possible preimages of $\overline{z}$. The two preimages can be found easily and therefore the relative index of y is easy to construct. In the most general setting,* *however, we encounter two problems: Firstly, a malicious* $\mathcal{R}$ *may be able to force the existence of only a single preimage of* $\overline{z}$ *w.r.t.* $\overline{h}$ that lies in L. *Secondly, it may be difficult to find the preimages of* $\overline{z}$ *that lie in* L.

We note that assuming the existence of one-way functions, in several cryptographic applications of interactive hashing (e.g., statistically-hiding bit-commitment, see [7, Theorem 6.1]) any protocol that is secure against an honest receiver can be complied into a protocol that is secure against a malicious receiver.

3.2 The Construction

Construction 3.7 (interactive hashing protocol)**.** *Let* $m(n) \in \mathbb{N}$ *and let* $\mathcal{H}$ *be a family of efficiently computable Boolean functions defined over strings of length* $\ell(n)$. *We define the interactive-hashing protocol* $(\mathcal{S}, \mathcal{R})$ *as follows:*

Common input: 1^n.
Sender's private input: $y \in \{0,1\}^{\ell(n)}$.

1. *For* $i = 1$ *to* $m(n)$*:*
    - (a) $\mathcal{R}$ *chooses uniformly at random* $h_i \in \mathcal{H}$ *and sends its description to* $\mathcal{S}$.
    - (b) $\mathcal{S}$ *sends* $z_i = h_i(y)$ *back to* $\mathcal{R}$.
2. $\mathcal{S}$ *locally outputs* y.
3. $\mathcal{R}$ *outputs* $(\overline{h}, \overline{z}) = ((h_1, \ldots, h_{m(n)}), (z_1, \ldots, z_{m(n)}))$.

The following lemma is immediate from Definitions 3.1 and 3.5.

Lemma 3.8. *Let* $\overline{\mathcal{H}}$ *be the* $m(n)$*-product-family of* $\mathcal{H}$*, then* $(\mathcal{S}, \mathcal{R})$ *is a secrecy-preserving w.r.t. semi-honest receiver* $\overline{\mathcal{H}}$*-interactive-hashing protocol.*

3.3 The Main Theorem - Binding

Theorem 3.9. *Let* W *be a binary relation and let* $L \subseteq \{0,1\}^{\ell(n)}$*. Let* $\mathcal{H}$ *be an efficient family of pairwise independent Boolean hash functions defined over string of length* $\ell(n)$*. Finally, let* $(\mathcal{S}, \mathcal{R})$ *be as in Construction 3.7 and let* A *be an algorithm trying to break the binding of* $(\mathcal{S}, \mathcal{R})$*. Then there exists an oracle algorithm* $M^{(\cdot)}$ *that given an oracle access to* A*, the following holds for large enough* n*:*

$$\Pr_{y \leftarrow L}[M^A(y) \in W_y] \in \Omega\left(\frac{2^{m'(n)}}{|L|} \cdot \frac{\varepsilon_A(n)^2}{n^8}\right) \ ,$$

where
$\varepsilon_A(n) \overset{def}{=} \Pr_{(o_A, o_{\mathcal{R}}) \leftarrow \langle A(1^n), \mathcal{R}(1^n) \rangle}[\mathsf{BindBreak}^{\mathsf{L,W}}(o_A, o_{\mathcal{R}})]$
and $m'(n) \overset{def}{=} \min\{m(n), \lfloor \log(|L|) \rfloor\}$.
Given that the oracle running is time T_A *and letting* $T_{\mathcal{H}}(n)$ *be an upper bound of the sampling and computing time of* $\mathcal{H}$*, then the running-time of* M^A *is* $\mathcal{O}(log(n)T_A(n) + m\log(n)T_{\mathcal{H}}(n))$.

Remark 3.10. *We point out that* M^A *does not need to know* L*,* W *or* ε_A.

Remark 3.11 (comparing the parameters to [12] and [13])**.** *For* $L = \{0,1\}^n$ *and* $m = n - 1$*, the success probability of* M^A *is* $\Omega(\frac{\varepsilon_A(n)^2}{n^8})$*, where the running-time is still* $\mathcal{O}(log(n)T_A(n) + m\log(n)T_{\mathcal{H}}(n))$*. We point that the same success probability and running-time apply also for the NOVY protocol (see Section 4 for details). This is an improvement in parameters compared with the analysis in [13, Lemma B.2]. There the algorithm runs in time* $\mathcal{O}(nT_A(n) + mnT_{\mathcal{H}}(n))$ *and breaks* f *with probability* $\Omega(\frac{\varepsilon_A(n)^3}{n^6})$*. Finally, in the [12, [Lemma 2] analysis, the algorithm runs in time* $\mathcal{O}(nT_A(n) + mnT_{\mathcal{H}}(n))$ *(same as in [13]) and only guarantees to break* f *with probability* $\Omega(\frac{\varepsilon_A(n)^{10}}{n^8})$.

The following corollaries follow Lemma 3.8 and Theorem 3.9.

Definition 3.12. *Let* W *be a relation and let* $L \subseteq \{0,1\}^{\ell(n)}$*. We say that* W *is hard-to-satisfy on* L *if for any* PPT A *the probability* $\Pr_{y \leftarrow L}[A(y) \in W_y]$ *is negligible in* n*.*

Corollary 3.13. *Let* L*,* m*,* $\overline{\mathcal{H}}$*,* W *and* $(\mathcal{S}, \mathcal{R})$ *be as in Theorem 3.9. If* W *is hard-to-satisfy on* L *and* $m > log(|L|) - \mathcal{O}(log(n))$*, then the protocol* $(\mathcal{S}, \mathcal{R})$ *is a binding and secrecy-preserving (w.r.t. semi-honest receiver)* $\overline{\mathcal{H}}$*-interactive hashing protocol w.r.t.* L *and* W*.*

Definition 3.14. *Let* $f : \{0,1\}^n \mapsto \{0,1\}^{\ell(n)}$ *be an efficiently computable function and let* $L \subseteq \{0,1\}^{\ell(n)}$*. We say that* f *is hard to invert over* L *if for any* PPT A *the probability* $\Pr_{y \leftarrow L}[A(y) \in f^{-1}(y)]$ *is negligible in* n*.*

Corollary 3.15. *Let* L*,* m*,* $\overline{\mathcal{H}}$ *and* $(\mathcal{S}, \mathcal{R})$ *be as in Theorem 3.9. Let* f *be hard to invert over* L*, and let* W *be the binary relation defined by* f *(i.e.,* $(x, y) \in W$ *iff* $f(x) = y$*). If* $m > \log(|L|) - \mathcal{O}(log(n))$*, then the protocol* $(\mathcal{S}, \mathcal{R})$ *is a computationally-binding secrecy-preserving (w.r.t. semi-honest receiver)* $\overline{\mathcal{H}}$*-interactive-hashing protocol w.r.t.* L *and* W*.*

The Proof of Theorem 3.9. For simplicity we drop the dependency on n whenever it is clear from the context. We assume w.l.o.g. that $m' = m$, since any adversary that violates the binding of the m-round protocol, can violate with the same probability the binding of the protocol with $m' < m$ rounds. We denote by A^r the restriction of A to some fixed random coins $r \in \{0,1\}^{T_A}$ and use throughout the proof the following random variables: For $k \in [m]$ and $(r, \overline{h^s}) \in$

$\{0,1\}^{T_A} \times \mathcal{H}^{\times k}$, let $A^{Com}(r, \overline{h^s}) \in \{0,1\}^k$ be A^r's answers when questioned by $\overline{h^s}$ and let $\mathsf{Consist}(r, \overline{h^s}) = \{y \in L : \forall i \in [k]\ \overline{h^s}_i(y) = A^{Com}(r, \overline{h^s})_i\}$ (i.e., the set of y's that are consistent with A's answers). Finally, we assume w.l.o.g. that for any sequence of questions $\overline{h} \in \mathcal{H}^{\times m}$, A^r outputs two pairs of elements $(x_0, y_0), (x_1, y_1) \in \{0,1\}^* \times \{0,1\}^{\ell(n)}$ and denote them by $A^{Dec}(r, \overline{h})$. For some value of $\texttt{ofs} \in \{0, \ldots, m-1\}$ that will be specified below, we consider the following algorithm for satisfying W on L:

```
$M^A(y)$:
  1. Choose uniformly at random $r \in \{0,1\}^{T_A}$.
  2. Let $\overline{h^s} \leftarrow Searcher(r, y)$.
  3. Return $Inverter(r, \overline{h^s})$.
```

where the algorithms $Searcher$ and $Inverter$ are defined as follows:

```
$Searcher(r, y)$:
  1. Fix $A$'s random coins to $r$.
  2. For $k = 1$ to $m - \texttt{ofs}$:
     Do the following $2\log(n)$ times:
       (a) Set a value for $h_k$ uniformly at random in $\mathcal{H}$.
       (b) If $A^{Com}(r, (h_1, \ldots, h_k))_k = h_k(y)$, break
           the inner loop.
  3. Return $(h_1, \ldots, h_{m-\texttt{ofs}})$.
```

```
$Inverter(r, \overline{h^s})$:
  1. Fix $A$'s random coins to $r$.
  2. Choose uniformly at random $\overline{h^e} \in \mathcal{H}^{\times \texttt{ofs}}$.
  3. Set $((x_0, y_0), (x_1, y_1)) \leftarrow A^{Dec}(r, (\overline{h^s}, \overline{h^e}))$.
  4. Return $x_0$ with probability half and $x_1$ otherwise.
```

Remark 3.16. *The value $\texttt{ofs}$ will depend in our proof on ε_A. This seems to contradict Remark 3.10 that M^A does not need to know ε_A. Nevertheless, $\texttt{ofs}$ can instead be selected at random with only a factor m decrease in the success probability of M^A. More interestingly, setting $\texttt{ofs} = 0$ will also guarantee M^A the succuss probability claimed in the theorem. The only affect of decreasing $\texttt{ofs}$ to zero is that $\overline{h^e}$ will be selected by the rewinding method of $Searcher$ rather than uniformly at random by $Inverter$. For every value $\overline{h^e}$ that satisfies $y \in \mathsf{Consist}(r, (\overline{h^s}, \overline{h^e}))$, we have that the probability of selecting it with the rewinding technique is only larger than the probability of uniformly selecting it. A value of $\overline{h^e}$ such that $y \notin \mathsf{Consist}(r, (\overline{h^s}, \overline{h^e}))$ will not contribute in our analysis to the success probability of M^A.*

It follows that the distinction between $Searcher$ and $Inverter$ is not necessary for the proof. Still, following [12], we find this distinction very useful for pedagogical reasons.

Assuming that we use the proper data structure to support the rewinding action, it follows that the running time of M^A is $\mathcal{O}(log(n)T_A(n) + m\log(n)T_{\mathcal{H}}(n))$. As a first step in proving correctness, we show that $\Pr_{y \leftarrow L}[M^A(y) \in W_y] \in \Omega\left(\frac{2^{m(n)}}{|L|} \cdot \frac{\varepsilon_A(n)^3}{n^6}\right)$, since this proof has somewhat nicer abstraction than the one proving the stronger bound claimed in the theorem. In Section 3.7, we present the modifications needed for the stronger result.

We would like to set the value of $\texttt{ofs}$ to $\lceil 6\log(n) + 2log(\frac{1}{\varepsilon_A})\rceil + C$, where $C \in \mathbb{N}$ is some universal constant determined by the analysis. For that we need to assume that $m > \lceil 6\log(n) + 2log(\frac{1}{\varepsilon_A})\rceil + C$. If $m \leq 6\log(n) + 2log(\frac{1}{\varepsilon_A}) + C$, then we can set $\texttt{ofs} = m$ and conclude the proof of the theorem directly as follows:

$$
\begin{aligned}
&\Pr_{y \leftarrow L}[M^A(y) \in W_y] && (1)\\
&= \sum_{y \in L} \frac{1}{|L|} \cdot \Pr_{(r,\overline{h}) \leftarrow \{0,1\}^{T_A} \times \mathcal{H}^{\times m}}[Inverter(r, \overline{h}) \in W_y]\\
&\geq \frac{1}{|L|} \sum_{y \in L} \frac{1}{2} \cdot \Pr\left[x_0 \in W_y \bigvee x_1 \in W_y\right]\\
&\geq \frac{\varepsilon_A}{|L|} \in \Omega\left(\frac{2^m}{|L|} \cdot \frac{\varepsilon_A^3}{n^6}\right) \quad ,
\end{aligned}
$$

where the last probability is taken over $(r, \overline{h}) \leftarrow \{0,1\}^{T_A} \times \mathcal{H}^{\times m}$ and $((x_0, y_0), (x_1, y_1)) \leftarrow A^{Dec}(r, \overline{h})$. We conclude that we can set $\texttt{ofs} = \lceil 6\log(n) + 2log(\frac{1}{\varepsilon_A})\rceil + C$, and assume that $m > \texttt{ofs}$.

Consider the following two distributions:

Definition 3.17.

- $D_{Ideal} \overset{def}{=} \left(r, \overline{h}, y\right)_{r \leftarrow \{0,1\}^{T_A}, \overline{h} \leftarrow \mathcal{H}^{\times(m-\texttt{ofs})}, y \leftarrow \mathsf{Consist}(r, \overline{h})}$
- $D_{Real} \overset{def}{=} \left(r, \overline{h}, y\right)_{r \leftarrow \{0,1\}^{T_A}, y \leftarrow L, \overline{h} \leftarrow Searcher(y, r)}$

Given that y is uniformly chosen in L, then D_{Real} is the distribution that $Inverter$ is invoked upon through the execution of M^A. Thus, the probability that $Inverter$ satisfies W over D_{Real} equals to the success probability of M^A. On the other hand, it is rather easy to show that the probability that $Inverter$ satisfies W over D_{Ideal} is noticeable (as a function of ε_A). Intuitively, this is because the distribution of $\overline{h}$ in D_{Ideal} is uniform and this is also the distribution of $\overline{h}$ that A encounters when interacting with $\mathcal{R}$. In fact, Lemma 3.20 states that the probability that $Inverter$ satisfies W over D_{Ideal} is well spread: Even if we ignore the contribution to the success probability of some sufficiently small number of values in the support of D_{Ideal}, this success probability will remain noticeable. To sum up, we know that $Inverter$ does well in satisfying W over D_{Ideal} and our goal is to show that it also does well over D_{Real}.

To this end, Lemma 3.27 will show that the distributions D_{Ideal} and D_{Real} are "not too far" from each other (in a sense defined below). This will indeed allow us to complete our proof.

3.4 Most Pairs Are Balanced

We start the formal proof by showing that in each step of the protocol the number of elements inside L that are consistent with the transcript so far is w.h.p. (regardless of A's answers) not faraway from the expected value.

Definition 3.18. *For $i \in \{0, \ldots, m\}$, the pair $(r, \overline{h}) \in \{0,1\}^{T_A} \times \mathcal{H}^{\times i}$ is* balanced *if*

$$\frac{|L|}{3 \cdot 2^i} \leq \left|\mathsf{Consist}(r, \overline{h})\right| \leq \frac{3 \cdot |L|}{2^i} \quad .$$

Claim 3.19. *For every $i \in \{0, \ldots, m\}$,*

$$\Pr_{(r,\overline{h}) \leftarrow \{0,1\}^{T_A} \times \mathcal{H}^{\times i}}[(r, \overline{h}) \text{ is balanced}] \geq 1 - \tfrac{6n^2 2^i}{|L|} \quad .$$

Proof. Omitted.

□

3.5 Analyzing the Success Probability of $Inverter$ on D_{Ideal}

As mentioned above, it is rather easy to prove (much like the proof for the case that $m = \texttt{ofs}$) that the success probability of $Inverter$ over D_{Ideal} is at least $\Omega(\frac{2^m}{|L|} \cdot \frac{\varepsilon_A^2}{n^6})$. Proving that, however, does not suffice to deduce that the success probability of $Inverter$ over D_{Real} is also high. The reason is that potentially the success probability of $Inverter$ over D_{Ideal} could stem from a relatively few elements that have significantly smaller probability mass w.r.t. D_{Real} than w.r.t. D_{Ideal}. To overcome this problem, we prove that the success of $Inverter$ is sufficiently high even if we flatten this probability such that the contribution of any single element is small. Having that, we are guaranteed that the success probability of $Inverter$ is high w.r.t. any distribution that assigns about the same mass to *most* elements in $sup(D_{Ideal})$ (and as we show later, D_{Real} satisfies this property). More formally, for every $(r, \overline{h^s}) \in \{0,1\}^{T_A} \times \mathcal{H}^{\times(m-\texttt{ofs})}$ we let

$$\varepsilon_{r,\overline{h^s}} \overset{\text{def}}{=} \Pr[\mathsf{BindBreak}^{\mathsf{L},\mathsf{W}}(o_A, o_{\mathcal{R}}) \mid \overline{h}_{1,\ldots,m-\texttt{ofs}} = \overline{h^s}] \ ,$$

where the probability is over $(o_A, o_{\mathcal{R}} = (\overline{h}, *)) \leftarrow \langle A^r(1^n), \mathcal{R}(1^n)\rangle$. That is, $\varepsilon_{r,\overline{h^s}}$ is the cheating probability of A conditioned on $(r, \overline{h^s})$. We define the *weight* of $y \in \mathsf{Consist}(r, \overline{h^s})$ by

$$w(r, \overline{h^s}, y) \overset{\text{def}}{=} \frac{1}{2} \Pr[\mathsf{BindBreak}^{\mathsf{L},\mathsf{W}}(o_A, o_{\mathcal{R}}) \bigwedge y \in \{y_0, y_1\} \,|\overline{h}_{1,\ldots,m-\texttt{ofs}} = \overline{h^s}],$$

where the probability is over $(o_A = ((*, y_0), (*, y_1)), o_{\mathcal{R}} = (\overline{h}, *)) \leftarrow \langle A^r(1^n), \mathcal{R}(1^n)\rangle$. Note that $w(r, \overline{h^s}, y)$ is a lower bound on the probability that $Inverter$ satisfies W on y conditioned on $(r, \overline{h^s})$. Also note that $\varepsilon_{r,\overline{h^s}} = \sum_{y \in \mathsf{Consist}(r,\overline{h^s})} w(r, \overline{h^s}, y)$. Finally, we define the *flattened weight* of $(r, \overline{h^s}, y)$ as

$$w_{\texttt{flt}}(r, \overline{h^s}, y) \overset{\text{def}}{=} \min\left\{\frac{\varepsilon_A}{2^{(C-1)/2} n^3}, w(r, \overline{h^s}, y)\right\} \quad ,$$

where $C > 0$ is the same universal constant that appears in the definition of $\texttt{ofs}$. The following lemma essentially says that the probability that $Inverter$ satisfies W over D_{Ideal} is well spread. That is, it does not come from satisfying some small number of heavy elements.

Lemma 3.20.

$$\mathsf{Ex}_{(r,\overline{h},y_s) \leftarrow D_{Ideal}}[w_{\texttt{flt}}(r, \overline{h^s}, y)] \in \Omega\left(\tfrac{2^m}{|L|} \cdot \tfrac{\varepsilon_A^3}{2^C n^6}\right) \quad .$$

Proof. Let $(r, \overline{h^s}) \in \{0,1\}^{T_A} \times \mathcal{H}^{\times(m-\texttt{ofs})}$. We assume for simplicity a non-increasing order on the elements of $\mathsf{Consist}(r, \overline{h^s})$ according to their wights and denote by $\mathsf{Consist}(r, \overline{h^s})_i$ the i^{th} element of $\mathsf{Consist}(r, \overline{h^s})$ by this order. The following claim states that the weight is not concentrated only on the first $\ell_{r,\overline{h^s}} \overset{\text{def}}{=} \lfloor\sqrt{2^{\texttt{ofs}-1}\varepsilon_{r,\overline{h^s}}}\rfloor$ heaviest elements of $\mathsf{Consist}(r, \overline{h^s})$.

Claim 3.21. $\sum_{i=\ell_{r,\overline{h^s}}+1}^{\left|\mathsf{Consist}(r,\overline{h^s})\right|} w(r, \overline{h^s}, \mathsf{Consist}(r, \overline{h^s})_i) \geq \varepsilon_{r,\overline{h^s}}/4$.

Proof. Let $Z = \bigcup_{i=1}^{\ell_{r,\overline{h^s}}} \left\{\mathsf{Consist}(r, \overline{h^s})_i\right\}$, by the pairwise independence of $\mathcal{H}$ it follows that,

$$\begin{aligned}
&\Pr_{(h_{m-\texttt{ofs}+1},\ldots,h_m) \leftarrow \mathcal{H}^{\times\texttt{ofs}}}[\exists y_0 \neq y_1 \in Z \text{ s.t.} \\
&\forall j \in \{m - \texttt{ofs} + 1, \ldots, m\} \ h_j(y_0) = h_j(y_1)] \\
&\leq \frac{|Z|^2}{2^{\texttt{ofs}}} \leq \frac{2^{\texttt{ofs}}\varepsilon_{r,\overline{h^s}}}{2 \cdot 2^{\texttt{ofs}}} = \varepsilon_{r,\overline{h^s}}/2 \ .
\end{aligned}$$

Recall that A cheats with probability $\varepsilon_{r,\overline{h^s}}$. Since the probability that both y_0 and y_1 it returns are inside Z is at most $\varepsilon_{r,\overline{h^s}}/2$, it follows that the probability that A^r cheats successfully while at least one of y_0 and y_1 is outside Z is at least $\varepsilon_{r,\overline{h^s}}/2$. Note that each event where A^r cheats successfully and outputs an element $y_i = y$, contributes half its probability to the total weight of y. Thus, the sum of weights of the elements inside $\mathsf{Consist}(r, \overline{h^s}) \setminus Z$ is at least $\varepsilon_{r,\overline{h^s}}/4$. □

Since $\varepsilon_{r,\overline{h^s}} = \sum_{y \in \mathsf{Consist}(r,\overline{h^s})} w(r, \overline{h^s}, y) \geq \sum_{i=1}^{\ell_{r,\overline{h^s}}} w\left(r, \overline{h^s}, \mathsf{Consist}(r, \overline{h^s})_i\right)$, it follows that $w(r, \overline{h^s}, \mathsf{Consist}(r, \overline{h^s})_{\ell_{r,\overline{h^s}}}) \leq \frac{\varepsilon_{r,\overline{h^s}}}{\ell_{r,\overline{h^s}}}$. Recall that

$\texttt{ofs} = \lceil 6\log(n) + 2log(\frac{1}{\varepsilon_A})\rceil + C$, therefore $w(r, \overline{h^s}, \mathsf{Consist}(r, \overline{h^s})_{\ell_{r,\overline{h^s}}}) \leq \frac{2\varepsilon_{r,\overline{h^s}}}{\ell_{r,\overline{h^s}}} = \frac{2\varepsilon_{r,\overline{h^s}}}{\lfloor\sqrt{2^{\texttt{ofs}-1}\varepsilon_{r,\overline{h^s}}}\rfloor} \leq \frac{\varepsilon_A}{2^{(C-1)/2}n^3}$ and thus for all $i > \ell_{r,\overline{h^s}}$, it holds that $w_{\texttt{flt}}(r, \overline{h^s}, \mathsf{Consist}(r, \overline{h^s})_i) = w(r, \overline{h^s}, \mathsf{Consist}(r, \overline{h^s})_i)$. Hence,

$$\sum_{y \in \mathsf{Consist}(r,\overline{h^s})} w_{\texttt{flt}}(r, \overline{h^s}, y) \tag{2}$$

$$\geq \sum_{i=\ell_{r,\overline{h^s}}+1}^{\left|\mathsf{Consist}(r,\overline{h^s})\right|} w(r, \overline{h^s}, \mathsf{Consist}(r, \overline{h^s})_i) \geq \varepsilon_{r,\overline{h^s}}/4 \ .$$

It follows that,

$$\begin{aligned}
&\mathop{\mathsf{Ex}}_{(r,\overline{h},y_s)\leftarrow D_{Ideal}} [w_{\texttt{flt}}(r, \overline{h^s}, y)] \\
= &\frac{1}{|\{0,1\}^{T_A} \times \mathcal{H}^{\times\texttt{ofs}}|} \\
&\cdot \sum_{(r,\overline{h^s},*)\in sup(D_{Ideal})} \frac{\sum_{y\in\mathsf{Consist}(r,\overline{h^s})} w_{\texttt{flt}}(r, \overline{h^s}, y)}{\left|\mathsf{Consist}(r, \overline{h^s})\right|} \\
\geq &\frac{1}{|\{0,1\}^{T_A} \times \mathcal{H}^{\times\texttt{ofs}}|} \cdot \sum_{(r,\overline{h^s})\in K} \frac{\varepsilon_{r,\overline{h^s}}}{4} / (\frac{3 \cdot |L|}{2^{m-\texttt{ofs}}}) \\
= &\frac{2^{m-\texttt{ofs}}}{12 \cdot |L|} \cdot \frac{1}{|\{0,1\}^{T_A} \times \mathcal{H}^{\times\texttt{ofs}}|} \sum_{(r,\overline{h^s})\in K} \varepsilon_{r,\overline{h^s}} \ ,
\end{aligned}$$

where $K \stackrel{\text{def}}{=} \{(r, \overline{h^s}) : \exists y \text{ s.t. } (r, \overline{h^s}, y) \in sup(D_{Ideal}) \bigwedge \left|\mathsf{Consist}(r, \overline{h^s})\right| \leq \frac{3\cdot|L|}{2^{m-\texttt{ofs}}}\}$. Claim 3.19 yields that $\Pr_{(r,\overline{h^s},*)\leftarrow D_{Ideal}}\left[\left|\mathsf{Consist}(r, \overline{h^s})\right| > \frac{3\cdot|L|}{2^{m-\texttt{ofs}}}\right] \in \mathcal{O}\left(\frac{n^2\cdot 2^{m-\texttt{ofs}}}{|L|}\right)$ and therefore,

$$\begin{aligned}
&\mathop{\mathsf{Ex}}_{(r,\overline{h},y_s)\leftarrow D_{Ideal}} [w_{\texttt{flt}}(r, \overline{h^s}, y)] \\
\geq &\frac{2^{m-\texttt{ofs}}}{12 \cdot |L|} \cdot \left(1 - \mathcal{O}(\frac{n^2 \cdot 2^{m-\texttt{ofs}}}{|L|})\right) \cdot \varepsilon_A \\
&\in \Omega\left(\frac{2^m}{|L|} \cdot \frac{\varepsilon_A^3}{2^C \cdot n^6}\right) \ .
\end{aligned}$$

□

3.6 Analyzing the Success Probability of $Inverter$ on D_{Real}

We would have liked to claim that D_{Ideal} is statistically close to D_{Real} and thus, the proof would immediately follow Lemma 3.20. Unfortunately, we can only prove that the two distributions are at statistical distance that is much bigger than the success probability of $Inverter$ on D_{Ideal}. Hence, we need to refine our approach by considering a different measure of distance. We call an element $y \in sup(D_{Ideal})$ "good", if $\frac{D_{Real}(y)}{D_{Ideal}(y)}$ is not too small. We prove that the total mass of the non-good elements in the support of D_{Ideal} is small. Thus, the probability that a good element is drawn from D_{Ideal} on which $Inverter$ does well is noticeable. Having that, we deduce that $Inverter$ also does well on D_{Real}. Let us turn to a more formal discussion.

The proximity measure.

Definition 3.22. *Let D_1 and D_2 be two distributions over a set Z, let $\varepsilon \in [0,1]$ and let $a \geq 1$. We say that D_1 (ε, a)-* approximates *D_2, if there exists a set $Z' \subseteq Z$ such that the following hold:*

1. *$D_1(Z') \leq \varepsilon$,*
2. *For every $x \in sup(D_1) \backslash Z'$ it holds that $\frac{1}{a} \leq \frac{D_1(x)}{D_2(x)} \leq a$.* [10]

The following propositions show that the proximity measure "behaves" similarly to the standard statistical distance measure. Since the proofs of following propositions are rather immediate, they are omitted. The first proposition enables us to use hybrid arguments when proving the proximity between distributions.

Proposition 3.23 (transitivity)**.** *Let D_1, D_2 and D_3 be distributions over a set Z, let $\varepsilon_1, \varepsilon_2 \in [0,1]$ and let $a_1, a_2 \geq 1$. Assuming that D_1 (ε_1, a_1)-approximates D_2 and that D_2 (ε_2, a_2)-approximates D_3, then D_1 $(\varepsilon_1 + a_1\varepsilon_2, a_1a_2)$-approximates D_3.*

Proposition 3.24 (average)**.** *Let $\left\{D_1^i\right\}_{i=1}^m$ and $\left\{D_2^i\right\}_{i=1}^m$ be two distributions ensembles over some set Z such that for every $i \in [m]$ it holds that D_1^i (ε_i, a)-approximates D_2^i. Let P be some distribution over $[m]$ and for every $j \in \{0,1\}$ let D_j be the distribution over Z defined as $D_j(x) = \sum_{i=1}^m P(i)D_j^i(x)$. Then D_1 (ε, a)-approximates D_2, where $\varepsilon = \mathsf{Ex}_{i\leftarrow P}[\varepsilon_i]$.*

Proposition 3.25 (extension)**.** *Let D_1 and D_2 be two distributions such that D_1 (ε, a)-approximates D_2 and let $g : \mathrm{sup}(D_1) \cup \mathrm{sup}(D_2) \mapsto \{0,1\}^*$ be a random process such that for every $x_0 \neq x_2 \in \mathrm{sup}(D_1) \cup \mathrm{sup}(D_2)$ it holds that $\Pr[g(x_0) = g(x_2)] = 0$, then $g(D_1)$ (ε, a)-approximates $g(D_2)$.*

The following proposition is where the usefulness of the new proximity measure lies, since it implies that the expected value of any predicate over two distributions that are close to each other is similar.

[10] Actually, for the purpose of this paper, it would suffice to require that $\frac{1}{a} \leq \frac{D_1(x)}{D_2(x)}$. We chose to use the symmetric definition since it seems more natural to us.

Proposition 3.26 (evaluation)**.** *Let D_1 and D_2 be two distributions such that D_1 (ε, a)-approximates D_2. Let $\delta > 0$ and let $g : \sup(D_1) \cup \sup(D_2) \mapsto [0, \delta]$, then $\mathsf{Ex}_{x \leftarrow D_2}[g(x)] \geq \frac{1}{a}(\mathsf{Ex}_{x \leftarrow D_1}[g(x)] - \varepsilon\delta)$.*

Lemma 3.27. *D_{Ideal} $\left(\mathcal{O}(\frac{2^m}{|L|} \cdot \frac{\varepsilon_A^2}{2^C \cdot n^3}), 81\right)$-approximates D_{Real}.*

Proof. We "bridge" between D_{Ideal} and D_{Real} using the following hybrid distributions. For every $\overline{h^s} \in \mathcal{H}^{\times k}$, we define the hybrid algorithm $Searcher^{\overline{h^s}}(r, y)$ that sets its first k hash functions to $\overline{h^s}$ and then continues as the original $Searcher$ algorithm does. For any $0 \leq k \leq m - \mathtt{ofs}$ let $D^k \stackrel{\text{def}}{=} \left(r, (\overline{h^s}, \overline{h^e}), y\right)$, where $r \leftarrow \{0,1\}^{T_A}, \overline{h^s} \leftarrow \mathcal{H}^{\times k}$, $y \leftarrow \mathsf{Consist}(r, \overline{h^s})$ and $\overline{h^e} \leftarrow Searcher^{\overline{h^s}}(r, y)_{k+1,\ldots,m-\mathtt{ofs}}$. Note that D^0 is equal to D_{Real} and that $D^{m-\mathtt{ofs}}$ is equal to D_{Ideal}. The following lemma states that every neighboring distributions are close to each other.

Lemma 3.28. *For every $0 \leq k < m - \mathtt{ofs}$, let $\ell_k = \frac{|L|}{3 \cdot 2^k}$ and let $\delta_k = \Pr_{(r,\overline{h^s}) \leftarrow \{0,1\}^{T_A} \times \mathcal{H}^{\times k}}[(r, \overline{h^s})$ is not balanced$]$. Then, D^{k+1} $\left(\delta_k + \frac{160 \cdot n^3}{\ell_k}, (1 + \frac{4}{n})\right)$-approximates D^k.*

Before proving Lemma 3.28, we use it to prove Lemma 3.27.

Proof. (of Lemma 3.27) By combining Lemma 3.28 and Proposition 3.23, we have that D_{Ideal} $\left(81 \sum_{k=0}^{m-\mathtt{ofs}-1}(\delta_k + \frac{160 \cdot n^3}{\ell_k}), 81\right)$-approximates D_{Real}. Claim 3.19 yields that $\delta_k < \frac{2n^2}{\ell_k}$ and therefore $\sum_{k=0}^{m-\mathtt{ofs}-1}(\delta_k + \frac{160 \cdot n^3}{\ell_k}) \in \mathcal{O}(\frac{n^3}{\ell_{m-\mathtt{ofs}}})$. We conclude that, D_{Ideal} $\left(\mathcal{O}(\frac{2^m}{|L|} \cdot \frac{\varepsilon_A^2}{2^C n^3}), 81\right)$-approximates D_{Real}. □

Proof. (of Lemma 3.28) Note that the only difference between D^k and D^{k+1} is their method of selecting y and $\overline{h}_{k+1}$. Therefore, in the proof we concentrate on the induced distributions on these values only. For any $(r, \overline{h^s}) \in \{0,1\}^{T_A} \times \mathcal{H}^{\times k}$, we define

- $D^0_{r,\overline{h^s}} \stackrel{\text{def}}{=} (y, h)_{y \leftarrow \mathsf{Consist}(r,\overline{h^s}), h \leftarrow Searcher^{\overline{h^s}}(r,y)_{k+1}}$
- $D^1_{r,\overline{h^s}} \stackrel{\text{def}}{=} (y, h)_{h \leftarrow \mathcal{H}, y \leftarrow \mathsf{Consist}(r,(\overline{h^s},h))}$

The proof of Lemma 3.28 will follow from the next Lemma.

Lemma 3.29. *Let $(r, \overline{h^s}) \in \{0,1\}^{T_A} \times \mathcal{H}^{\times k}$ be balanced, then $D^1_{r,\overline{h^s}}$ $(\frac{160 \cdot n^3}{\ell_k}, (1 + \frac{4}{n}))$-approximates $D^0_{t,\overline{h^s}}$.*

To see that the proof of Lemma 3.28 indeed follows by the above one, note that by the extension and average properties of the proximity measure (Propositions 3.25 and 3.24) and Lemma 3.29, it follows that conditioned on any balanced $(r, \overline{h^s})$, the distribution D^{k+1} $\left(\frac{160 \cdot n^3}{\ell_k}, (1 + \frac{4}{n})\right)$-approximates D^k. Since δ_k is the probability that $(r, \overline{h^s})$ is not balanced, the proof of Lemma 3.28 follows.

Proof. (of Lemma 3.29) Consider the Boolean matrix $T^{\left|\mathsf{Consist}(r,\overline{h^s})\right| \times |\mathcal{H}|}$, where $T(y, h) = 1$ iff $A^{Com}(r, (\overline{h^s}, h))_{k+1} = h(y)$ and zero otherwise. We identify the indices into T with the set $\mathsf{Consist}(r, \overline{h^s}) \times \mathcal{H}$. The distribution $D^1_{r,\overline{h^s}}$ can be described in relation to T as follows: Choose a random column of T and draw the index of a random one entry from this column (where a "one entry" is simply an entry of the matrix that is assigned the value one). The distribution $D^0_{t,\overline{h^s}}$ can also be described in relation to T as follows: Choose a random row of T and for $2\log(n)$ times draw a random entry from this raw. If a one entry is drawn, then choose its index and stop drawing, otherwise select the index of the last drawn entry.

Let us start with an informal discussion. Compare the matrix T with the matrix $\widehat{T}^{\left|\mathsf{Consist}(r,\overline{h^s})\right| \times |\mathcal{H}|}$, where $\widehat{T}(y, h) = h(y)$. Note that T can be derived from $\widehat{T}$ by flipping all values in some of its columns (where the column which corresponds to h is flipped whenever $A^{Com}(r, (\overline{h^s}, h))_{k+1} = 0$). By the pairwise independence of $\mathcal{H}$, it follows that most *columns* of $\widehat{T}$ are balanced (have about the same number of zeros and ones) and thus the same holds for T. Hence, the mass that $D^1_{r,\overline{h^s}}$ assigns to most of the one entries of T is close to $\frac{1}{|\mathcal{H}|} \cdot \frac{2}{\left|\mathsf{Consist}(r,\overline{h^s})\right|}$. Using again the pairwise independent of $\mathcal{H}$, we can prove that most *rows* of T are balanced. Hence, the mass that $D^0_{r,\overline{h^s}}$ assigns to most one entries in T is also close to $\frac{1}{|\mathcal{H}|} \cdot \frac{2}{\left|\mathsf{Consist}(r,\overline{h^s})\right|}$. Since the support of $D^1_{r,\overline{h^s}}$ and the indices set of one entries in T are the same, the proof of the Lemma 3.29 follows.

Let us turn to the formal proof. We define $\mathcal{H}^{Bad} \stackrel{\text{def}}{=} \{h \in \mathcal{H} : \Pr_{y \leftarrow \mathsf{Consist}(r,\overline{h^s})}[T(h, y) = 1] \notin [\frac{1}{2} \cdot (1 - \frac{1}{n}), \frac{1}{2} \cdot (1 + \frac{1}{n})]\}$. The following claim, whose proof is immediate by the pairwise independence of $\mathcal{H}$ (see Lemma 2.2), states that the relative size of $\mathcal{H}^{Bad}$ is small.

Claim 3.30. $\Pr_{h \leftarrow \mathcal{H}}[h \in \mathcal{H}^{Bad}] \leq \frac{2n^2}{\ell_k}$.

Similarly, we define $Y^{Bad} \stackrel{\text{def}}{=} \{y \in \mathsf{Consist}(r, \overline{h^s}) : \Pr_{h \leftarrow \mathcal{H}}[T(h, y) = 1] \notin [\frac{1}{2} \cdot (1 - \frac{1}{n}), \frac{1}{2} \cdot (1 + \frac{1}{n})]\}$. The following claim states the size of Y^{Bad} is small.

Claim 3.31. $\left|Y^{Bad}\right| < 54n^3$.

Proof. Let $Y^{Bad}_{Law} \overset{\text{def}}{=} \{y \in \mathsf{Consist}(r, \overline{h^s}) : \Pr_{h\leftarrow\mathcal{H}}[T(h,y) = 1] < \frac{1}{2} \cdot (1 - \frac{1}{n})\}$. We assume that $\left|Y^{Bad}_{Law}\right| > 27n^3$ and derive a contradiction (the proof that $\left|Y^{Bad} \backslash Y^{Bad}_{Law}\right| < 27n^3$ is analogous). Consider the matrix $T|_{Y^{Bad}_{Law}}$, the restriction of T to the rows Y^{Bad}_{Law}. By definition, the rows of $T|_{Y^{Bad}_{Law}}$ have more zeros than ones. Hence, the matrix $T|_{Y^{Bad}_{Law}}$ itself has more zeros than ones. On the other hand, by the pairwise independence of $\mathcal{H}$ it follows that most columns of $T|_{Y^{Bad}_{Law}}$ are balanced (have about the same number of zeros of ones). Therefore, $T|_{Y^{Bad}_{Law}}$ itself is balanced and a contradiction is derived. More formally, for $h \in \mathcal{H}$ let T_h be the number of ones in the h column, that is $T_h = \sum_{y \in Y^{Bad}_{Law}} T(y,h)$. We upper bound the expectation of T_h as follows,

$$\underset{h\leftarrow\mathcal{H}}{\mathsf{Ex}}[T_h] = \underset{h\leftarrow\mathcal{H}}{\mathsf{Ex}}[\sum_{y \in Y^{Bad}_{Law}} T(y,h)] \tag{3}$$

$$= \sum_{y \in Y^{Bad}_{Law}} \underset{h\leftarrow\mathcal{H}}{\mathsf{Ex}}[T(y,h)] < \left|Y^{Bad}_{Law}\right| (\frac{1}{2} - \frac{1}{2n}) \ .$$

Recall that $T(h,y) = 1$ if $A^{Com}(r, (\overline{h^s}, h))_{k+1} = h(y)$ and zero otherwise. Since the set Y^{Bad} is large enough, Lemma 2.2 yields that a random h splits w.h.p. the elements of Y^{Bad} into two almost equals size according to their consistency with A's answer on h. That it, $\Pr_{h\leftarrow\mathcal{H}}\left[T_h < \left|Y^{Bad}_{Law}\right| \cdot (\frac{1}{2} - \frac{1}{3n})\right] < \frac{9n^2}{\left|Y^{Bad}_{Law}\right|} \leq \frac{1}{3n}$. Thus,

$$\underset{h\leftarrow\mathcal{H}}{\mathsf{Ex}}[T_h] \tag{4}$$

$$\geq \frac{1}{|H|} \cdot \sum_{h \in \mathcal{H} : T_h \geq \left|Y^{Bad}_{Law}\right| \cdot (\frac{1}{2} - \frac{1}{3n})} \left|Y^{Bad}_{Law}\right| \cdot (\frac{1}{2} - \frac{1}{3n})$$

$$> (1 - \frac{1}{3n}) \cdot \left|Y^{Bad}_{Law}\right| \cdot (\frac{1}{2} - \frac{1}{3n})$$

$$> \left|Y^{Bad}_{Law}\right| \cdot (\frac{1}{2} - \frac{1}{2n}) \ ,$$

and a contradiction is derived. □

The next claim concludes the proof of Lemma 3.29 by presenting a set that actualizes the stated proximity distance between $D^1_{r,\overline{h^s}}$ and $D^0_{r,\overline{h^s}}$.

Claim 3.32. *Let* $Z \overset{def}{=} \left\{(y,h) \in const(r, \overline{h^s}) \times \mathcal{H} : y \in Y^{Bad} \bigvee h \in \mathcal{H}^{Bad}\right\}$, *then*

1. $D^1_{r,\overline{h^s}}(Z) \leq \frac{160 \cdot n^3}{\ell_k}$.
2. *For every* $(y,h) \in sup(D^1_{r,\overline{h^s}}) \backslash Z$, *it holds that* $\frac{D^1_{r,\overline{h^s}}(y,h)}{D^0_{r,\overline{h^s}}(y,h)} \in [\frac{1}{1+\frac{4}{n}}, 1 + \frac{4}{n}]$.

Proving Claim 3.32.1. Consider the partitioning of Z into $Z_1 \overset{\text{def}}{=} const(r, \overline{h^s}) \times \mathcal{H}^{Bad}$ and $Z_2 \overset{\text{def}}{=} Y^{Bad} \times (\mathcal{H} \backslash \mathcal{H}^{Bad})$. Since $\Pr_{D^1_{r,\overline{h^s}}}[Z_1] \leq \Pr_{h\leftarrow\mathcal{H}}[\mathcal{H}^{Bad}]$, Claim 3.30 yields that $\Pr_{D^1_{r,\overline{h^s}}}[Z_1] \leq \frac{2n^2}{\ell_k}$. Claim 3.31 yields that for any $h \in \mathcal{H}$, it holds that $\Pr_{(y,h')\leftarrow D^1_{r,\overline{h^s}}}[y \in Y^{Bad} \mid h' = h] \leq \frac{54n^3}{\left|\mathsf{Consist}(r,(\overline{h^s},h))\right|}$. Recall that for any $h \in \mathcal{H} \backslash \mathcal{H}^{Bad}$, it holds that $\left|\mathsf{Consist}(r,(\overline{h^s},h))\right| > (1 - \frac{1}{n}) \frac{\left|\mathsf{Consist}(r,\overline{h^s})\right|}{2}$. Since $(r, \overline{h^s})$ is balanced, it follows that $\left|\mathsf{Consist}(r,(\overline{h^s},h))\right| > (1 - \frac{1}{n}) \frac{\ell_k}{2}$. Thus, for any $h \in \mathcal{H} \backslash \mathcal{H}^{Bad}$, it holds that $\Pr_{(y,h')\leftarrow D^1_{r,\overline{h^s}}}[y \in Y^{Bad} \mid h' = h] \leq \frac{1}{(1-\frac{1}{n})} \cdot \frac{2 \cdot 54n^3}{\ell_k} \leq \frac{150n^3}{\ell_k}$ and therefore, $\Pr_{D^1_{r,\overline{h^s}}}[Z_2] \leq \frac{150n^3}{\ell_k}$. We conclude that $\Pr_{D^1_{r,\overline{h^s}}}[Z] = \Pr_{D^1_{r,\overline{h^s}}}[Z_1] + \Pr_{D^1_{r,\overline{h^s}}}[Z_2] \leq \frac{160 \cdot n^3}{\ell_k}$. □

Proving Claim 3.32.2. For $(y,h) \in sup(D^1_{r,\overline{h^s}}) \backslash Z$, it holds that $D^1_{r,\overline{h^s}}(y,h) = \Pr_{D^1_{r,\overline{h^s}}}[h] \cdot \Pr_{D^1_{r,\overline{h^s}}}[y \mid h] = \frac{1}{|\mathcal{H}|} \cdot \frac{1}{\left|\left\{y' \in \mathsf{Consist}(r,\overline{h^s}) : T(y',h)=1\right\}\right|}$. Since $h \notin \mathcal{H}^{Bad}$, it follows that $D^1_{r,\overline{h^s}}(y,h) \in [\frac{\gamma}{1+\frac{1}{n}}, \frac{\gamma}{1-\frac{1}{n}}]$, where $\gamma = \frac{1}{|\mathcal{H}|} \cdot \frac{2}{\left|\mathsf{Consist}(r,\overline{h^s})\right|}$. Similarly, we have that $D^0_{r,\overline{h^s}}(y,h) = \frac{1}{\left|\mathsf{Consist}(r,\overline{h^s})\right|} \cdot \Pr_{D^0_{r,\overline{h^s}}}[h \mid y]$. Calculating the value of $\Pr_{D^0_{r,\overline{h^s}}}[h \mid y]$, however, is a bit more subtle. Conditioned on y, it might be that the entry drawn from T is zero. Thus, the conditional probability of h is not the uniform one over $\{h' \in \mathcal{H} : T(y,h') = 1\}$. Still, since $y \notin Y^{Bad}$, we have that the conditional probability that $T(y,h') \neq 1$ is in $o(\frac{1}{n})$. (Recall the description $D^0_{r,\overline{h^s}}$: For $2\log(n)$ rounds a random entry is selected from the y row of T, only if all the selected entries are zeros, then a zero entry is chosen). Therefore, the conditional probability of h is close to uniform over $\{h' \in \mathcal{H} : T(y,h') = 1\}$ and thus, $D^0_{r,\overline{h^s}}(y,h) = \frac{1}{\left|\mathsf{Consist}(r,\overline{h^s})\right|} \cdot \frac{1}{|\{h' \in \mathcal{H} : T(y,h')=1\}|} \cdot (1 \pm o(\frac{1}{n}))$. Using again the fact that $y \notin Y^{Bad}$, it follows that $D^0_{r,\overline{h^s}}(y,h) \in [\frac{\gamma - o(\frac{1}{n})}{1+\frac{1}{n}}, \frac{\gamma + o(\frac{1}{n})}{1-\frac{1}{n}}]$ and we conclude that $\frac{D^1_{r,\overline{h^s}}(y,h)}{D^0_{r,\overline{h^s}}(y,h)} \in [\frac{1}{1+\frac{4}{n}}, 1 + \frac{4}{n}]$. □

Putting it all together

Recall that $\Pr[Inverter(r, \overline{h}) \in W_y] \geq w(r, \overline{h}, y)$. We therefore have that $\Pr_{y\leftarrow L}[M^A(y) \in W_y] = \Pr_{(r,\overline{h},y)\leftarrow D_{Real}}[Inverter(r, \overline{h}) \in W_y] \geq \mathsf{Ex}_{(r,\overline{h},y)\leftarrow D_{Real}}[w_{\texttt{flt}}(r, \overline{h}, y)]$. We can now relate this expectation to an expectation over the distribution D_{Ideal} (on which we have a better handle). For that we use Lemma 3.27 regarding the proximity of the distributions

D_{Real} and D_{Ideal} and the evaluation property of the proximity measure (Proposition 3.26). Recalling that by its definition $w_{\mathtt{flt}}(r,\overline{h},y) \leq \frac{\varepsilon_A}{2^{C/2}n^3}$, we can deduce that

$$\begin{aligned}&\Pr_{y\leftarrow L}[M^A(y) \in W_y] \\ &\geq \frac{1}{81}\Bigg(\underset{(r,\overline{h},y)\leftarrow D_{Ideal}}{\mathsf{Ex}}[w_{\mathtt{flt}}(r,\overline{h},y)] \\ &-\mathcal{O}((\frac{2^m}{|L|}\cdot\frac{\varepsilon_A^2}{2^C\cdot n^3})\cdot\frac{\varepsilon_A}{2^{C/2}\cdot n^3})\Bigg) \ .\end{aligned}$$

Finally, Lemma 3.20 yields that,

$$\begin{aligned}&\Pr_{y\leftarrow L}[M^A(y) \in W_y] \\ \geq\ &\frac{1}{81}\left(\Omega(\frac{2^m}{|L|}\cdot\frac{\varepsilon_A^3}{2^C\cdot n^6}) - \mathcal{O}(\frac{2^m}{|L|}\cdot\frac{\varepsilon_A^3}{2^{\frac{3}{2}C}\cdot n^6})\right) \\ \in\ &\Omega(\frac{2^m}{|L|}\cdot\frac{\varepsilon_A^3}{n^6}) \ ,\end{aligned}$$

for large enough C. □

3.7 Achieving $\Omega(\frac{2^m}{|L|}\cdot\frac{\varepsilon_A^2}{n^8})$.

Since the success probability of M^A is at most the success probability of $Inverter$ over D_{Ideal} (at least by our proof's method) and since the latter is at most $\frac{\varepsilon_A}{2^{\mathtt{ofs}}}$, in order to get a better bound on M^A's success probability, we have to consider smaller values for $\mathtt{ofs}$. Following the same lines as the proof of Lemma 3.27, we could prove that for any value of $\mathtt{ofs}$, it holds that D_{Ideal} $\left(\mathcal{O}(\frac{2^m}{|L|}\cdot n^3\cdot 2^{-\mathtt{ofs}}, 81\right)$-approximates D_{Real} (where D_{Ideal} and D_{Real} are redefined w.r.t. the new value of $\mathtt{ofs}$.). The problem is, however, that the value we have previously chosen for $\mathtt{ofs}$ is the smallest value for which the "additive part" (i.e., the $\mathcal{O}(\frac{2^m}{|L|}\cdot n^3\cdot 2^{-\mathtt{ofs}})$ part) in the proximity gap between D_{Ideal} and D_{Real} does not overwhelm the success probability of $Inverter$ on D_{Ideal}.

Fortunately, by reexamining the proof of Lemma 3.27 w.r.t. $\mathtt{ofs} = \lceil 8\log(n) + \log(\frac{1}{\varepsilon_A})\rceil + 13$, we can prove that the set that actualizes the proximity between D_{Ideal} and D_{Real} is rather evenly spread along the possible pairs of $(r,\overline{h^s})$. Namely, we have the following lemma.

Lemma 3.33 (stronger version of Lemma 3.27)**.** *There exists a set $T \subseteq sup(D_{Ideal})$ such that the following hold:*

1. *For any $(r,\overline{h},y) \in sup(D_{Ideal})\backslash T$ it holds that* $\frac{1}{81} \leq \frac{D_{Real}(r,\overline{h},y)}{D_{Ideal}(r,\overline{h},y)} \leq 81$,
2. $\Pr_{(r,\overline{h},*)\leftarrow D_{Ideal}}[\left|\mathsf{Consist}(r,\overline{h^s})\cap T\right| > 54n^4] \in \mathcal{O}(\frac{\varepsilon_A}{n^5})$.

Proof. Omitted. □

The advantage of Lemma 3.33 over Lemma 3.27, is that it guarantees that conditioned on almost every choice of $(r,\overline{h^s})$, the distribution D_{Ideal} approximates D_{Real} well. In contrast to Lemma 3.27 that only guarantees that D_{Ideal} approximates D_{Real} well on the average over the choice of $(r,\overline{h^s})$. In particular, the above lemma allows us to relate the success probability of M^A over D_{Real} to the success probability of M^A over D_{Ideal} conditioned on, almost, all values of $(r,\overline{h^s})$.

Before using Lemma 3.27 to derive Theorem 3.9 (rather than the weaker version of the previous section), we first need to prove a different version of Lemma 3.20. Recall that Lemma 3.20 states that the success probability of $Inverter$ over D_{Ideal} is noticeable, even if we ignore elements on which $Inverter$'s success probability is higher than some threshold. The next lemma states that the success probability of $Inverter$ over D_{Ideal} is noticeable, even if we ignore elements of some given set, T, whose relative part in all but $\frac{\varepsilon_A}{2}$ of the pairs $(r,\overline{h^s})$ is not too big (keep in mind that the set of Lemma 3.33 is such a set).

Lemma 3.34 (new version of Lemma 3.20)**.** *Let $T \subseteq sup(D_{Ideal})$ and let $w_{\overline{T}}(x) = w(x)$ if $x \notin T$ and zero otherwise. If* $\Pr_{(r,\overline{h},*)\leftarrow D_{Ideal}}\left[\left|\mathsf{Consist}(r,\overline{h^s})\cap T\right| > 54n^4\right] < \varepsilon_A/2$, *then* $\mathsf{Ex}_{(r,\overline{h},y)\leftarrow D_{Ideal}}\left[w_{\overline{T}}(r,\overline{h^s},y)\right] \in \Omega(\frac{2^m}{|L|}\cdot\frac{\varepsilon_A^2}{n^8})$.

Proof. Omitted. □

Putting it all together

Let T be the set whose existence is guaranteed by Lemma 3.33. Recall that $\Pr[Inverter(r,\overline{h}) \in W_y] \geq w(r,\overline{h},y)$. We therefore have that $\Pr_{y\leftarrow L}[M^A(y) \in W_y] = \Pr_{X(r,\overline{h},y)\leftarrow D_{Real}}[Inverter(r,\overline{h}) \in W_y] \geq \mathsf{Ex}_{(r,\overline{h},y)\leftarrow D_{Real}}[w_{\overline{T}}(r,\overline{h},y)]$. Since D_{Ideal} approximates D_{Real} well on any element in T, it follows that

$$\Pr_{y\leftarrow L}[M^A(y) \in W_y] \geq \frac{1}{81}\underset{(r,\overline{h},y)\leftarrow D_{Ideal}}{\mathsf{Ex}}[w_{\overline{T}}(r,\overline{h},y)] \ .$$

Finally, Lemma 3.34 yields that,

$$\Pr_{y\leftarrow L}[M^A(y) \in W_y] \in \Omega(\frac{2^m}{|L|}\cdot\frac{\varepsilon_A^2}{n^8}) \ .$$

4 Applying Our New Proof to NOVY

The NOVY protocol is essentially an instance of the protocol given in Construction 3.7, where the number of rounds is set to $n-1$ and W is naturally defined by a one-way permutation (i.e., given a one-way permutation f over strings of length n, then W consists on all the pairs $(x, f(x))$ where $x \in \{0,1\}^n$). The difference is, however, that rather than using the same family of Boolean pairwise independent hash functions in each round, the NOVY protocol uses a different family for each round. Specifically, the protocol's i^{th} family $\mathcal{H}^i$ is defined by the set of all strings of the form $0^{i-1}1\{0,1\}^{n-i}$, where for $h \in \mathcal{H}^i$ and $x \in \{0,1\}^n$ the hash value $h(x)$ is defined as $\langle h, x \rangle \ mod \ 2$ (i.e., the inner-product of h and x modulo 2).

We would like to apply Theorem 3.9 also to the NOVY protocol. We could do so directly if the families $\{\mathcal{H}^i\}_{i=1}^{n-1}$ would be guaranteed to be pairwise independent w.r.t. $\{0,1\}^n$. [11] The latter, however, does not hold and therefore we have to refine our approach. Fortunately, the proof of the theorem does not require that the families of Boolean hash function to be pairwise independent w.r.t. the initial set of inputs L (in the NOVY case w.r.t. $\{0,1\}^n$), but rather to be pairwise independent w.r.t. the elements of the initial set that are consistent with the protocol so far. It turns out that given that the initial set is $\{0,1\}^n$, the families of Boolean hash functions used by NOVY are "enough" pairwise independent on the relevant set. Thus, the proof of Theorem 3.9 can be also applied to the NOVY setting.

Formal proof of the above discussion, appears in the full version of this paper.

5 Discussion and Further Research

One interesting question is to come with a reduction from interactive hashing to one-way permutation that is even more security preserving. Particularly, is there such a reduction that is linearly-preserving [9] (i.e., where the time-success ratio of an adversary inverting the one-way permutation is only larger by a multiplicative polynomial factor than the time-success ratio of an adversary breaking the interactive hashing protocol). There are three possible directions for an improvement: (1) Presenting a more secure protocol than the NOVY protocol (or our variant), (2) Giving a better reduction from an adversary that breaks the interactive hashing to one that breaks the one-way permutations, or (3) Improving the analysis of the reduction mentioned in (2).

Note that our improvement in parameters over the NOVY proof is mainly in the third item (i.e., the analysis of the reduction). In the full version of this paper, we prove that our analysis cannot be pushed much further. Namely, we present a (non-efficient) adversary A that breaks the binding of the NOVY protocol with probability ε, but M^A breaks the underlying one-way permutation with probability at most $2 \cdot \varepsilon^{1.4}$.

[11] Note that the proof of Theorem 3.9 does not require that the same family is used in each round.

Acknowledgments

The research of the second author is supported by grant no. 1300/05 from the Israel Science Foundation. We are grateful to Moni Naor and Ronen Shaltiel for helpful conversations. We are also grateful to the anonymous referees for their useful comments.

References

[1] G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *J. Computer and System Sciences*, 37(2).

[2] C. Cachin, C. Crépeau, and J. Marcil. Oblivious transfer with a memory-bound receiver.

[3] I. Carter and M. Wegman. Universal classes of hash functions. In *9th STOC*.

[4] C. Crpeau and G. Savvides. Optimal reductions between oblivious transfers using interactive hashing. In *Eurocrypt '06, LNCS*, 2006.

[5] Y. Ding, D. Harnik, R. Shaltiel, and A. Rosen. Constant round oblivious transfer in the bounded storage model. In *1st TCC*, 2004.

[6] O. Goldreich. Randomized methods in computation - lecture notes. 2001.

[7] I. Haitner, O. Horvitz, J. Katz, C. Koo, R. Morselli, and R. Shaltiel. Reducing complexity assumptions for statistically-hiding commitment. In *EUROCRYPT*.

[8] I. Haitner and O. Reingold. Statistically-hiding commitment from any one-way function, 2006.

[9] A. Herzberg and M. Luby. Pubic randomness in cryptography.

[10] T. Koshiba and Y. Seri. Round-efficient one-way permutation based perfectly concealing bit commitment scheme. ECCC, TR06-093, 2006.

[11] Y. Lindell. Parallel coin-tossing and constant-round secure two-party computation. *JCRYPTOLOGY*, 16(3).

[12] M. Naor, R. Ostrovsky, R. Venkatesan, and M. Yung. Perfect zero-knowledge arguments for NP using any one-way permutation. *JCRYPTOLOGY*, 11(2).

[13] M. Nguyen, S. Ong, and S. Vadhan. Statistical zero-knowledge arguments for NP from any one-way function. In *39th FOCS*, 2006.

[14] M. Nguyen and S. Vadhan. Zero knowledge with efficient provers. In *38th STOC*, 2006.

[15] R. Ostrovsky, R. Venkatesan, and M. Yung. Fair games against an all-powerful adversary, May 07 1991.

[16] R. Ostrovsky, R. Venkatesan, and M. Yung. Secure commitment against A powerful adversary. In *9th Annual Symposium on Theoretical Aspects of Computer Science*, volume 577 of *lncs*, Cachan, France, 13–15 Feb. 1992. Springer.

[17] R. Ostrovsky, R. Venkatesan, and M. Yung. Interactive hashing simplifies zero-knowledge protocol design. In *EUROCRYPT*, 1993.

[18] H. Wee. One-way permutations, interactive hashing and statistically hiding commitments. In *4th TCC*, 2007.